Journal of Educational Psychology
2013, Vol. 105, No. 4, 1126–1141

0022-0663/13/$12.00 DOI: 10.1037/a0032063

A Tutoring System That Simulates the Highly Interactive Nature of Human Tutoring

Sandra Katz and Patricia L. Albacete
University of Pittsburgh

For some time, it has been clear that students who are tutored generally learn more than students who experience classroom instruction (e.g., Bloom, 1984). Much research has been devoted to identifying features of tutorial dialogue that can explain its effectiveness, so that these features can be simulated in natural-language tutoring systems. One hypothesis is that the highly interactive nature of tutoring itself promotes learning—that is, the *interaction hypothesis*. Although reasonable and agreeing with much research, the interaction hypothesis raises the question of what linguistic mechanisms are involved: that is, which features of "highly interactive" dialogues trigger what processes that are conducive to learning? Our overall strategy in the research described in this article was to inform this question by identifying co-constructed discourse relations in tutorial dialogues whose frequency of occurrence predicts learning, identify the context in which these relations occur, and use this knowledge to formulate decision rules to guide automated dialogues. We used Rhetorical Structure Theory to identify and tag co-constructed discourse relations in a large corpus of physics tutoring dialogues. Our analyses suggest that the effectiveness of human tutoring might well lie in the language of tutoring itself. Moreover, the types of co-constructed discourse relations that predict learning seem to vary based on students' ability level. We describe Rimac, a natural-language tutoring system that implements an initial set of decision rules based on these analyses. These rules guide reflective dialogues about the concepts associated with physics problems. Rimac is being pilot tested in high school physics classes.

Keywords: instructional dialogue, natural-language tutoring systems, Rhetorical Structure Theory

Educators and policy makers in the United States have looked to educational technology as a tool to increase students' proficiency in math, science, reading, and other subject matter domains. For example, early in his administration, President Obama (2009) challenged developers of intelligent tutoring systems (ITSs) to develop "learning software as effective as a personal tutor" (para. 19). Apparently, Obama cast this challenge a bit too late. A recent meta-analysis of research comparing the effectiveness of human tutors with state-of-the-art ITSs showed that ITSs have already nearly caught up with human tutors (VanLehn, 2011), with effect sizes (*d*) of 0.76 for human tutoring and 0.79 for ITSs relative to no tutoring (e.g., problem solving and reading, without feedback).[1] This comparison raises the bar for developers of ITSs. The challenge now is to develop automated tutors that can perform even better than human tutors with learners of all types.

Several researchers have proposed that the large effect sizes of human tutoring can be attributed to its highly interactive nature—that is, the high degree to which the student and tutor respond to and build upon each other's dialogue moves (e.g., M. T. H. Chi, Siler, Jeong, Yamauchi, & Hausmann, 2001;[2] Graesser, Person, & Magliano, 1995; van de Sande & Greeno, 2010). However, an important line of research conducted in the past few years to test this so-called *interaction hypothesis* showed that it is neither how much interaction takes place during tutoring that is important, nor the granularity of interaction—for example, whether the student and tutor discuss a step toward solving a problem or the substeps that lead to that step. Instead, what matters most is how *well* the interaction is carried out—for example, what content is addressed and how it is addressed in a particular dialogue context (e.g., M. Chi, VanLehn, Litman, & Jordan, 2010, 2011a, 2011b; Murray & VanLehn, 2006).

This important finding suggests that the key to building tutoring systems that surpass the effectiveness of human tutors is to specify

This article was published Online First September 9, 2013.

Sandra Katz and Patricia L. Albacete, Learning Research and Development Center, University of Pittsburgh.

The authors thank the Rimac project team—Stefani Allegretti, Michael Ford, Pamela Jordan, Kevin Krost, Michael Lipschultz, Diane Litman, Tyler McConnell, Scott Silliman, Elizabeth Spiegel, Christine Wilson, and Peter Wu—for their contributions. The research reported here was supported by the Institute of Education Sciences, U.S. Department of Education, through Grant R305A10063 to the University of Pittsburgh. The opinions expressed are those of the authors and do not represent the views of the Institute or the U.S. Department of Education.

Correspondence concerning this article should be addressed to Sandra Katz, Learning, Research and Development Center, 3939 O'Hara Street, Pittsburgh, PA 15260. E-mail: katz@pitt.edu

[1] VanLehn's (2011) review showed that the two sigma effect for human tutoring reported by Bloom (1984) testifies to the importance of a mastery learning standard and is not typical of human tutoring in general.

[2] Two important players in the field of tutoring research have the same last name and first initial. In our citations, we use M. T. H. Chi to refer to Michelene ("Micki") T. H. Chi and M. Chi to refer to Min Chi.

what we mean by *effective interaction* and to formulate "policies for selecting the tutorial action at each microstep when there are multiple action options available" (M. Chi et al., 2011a, p. 87). Such "policies" have alternatively been called *pedagogical tutoring tactics* or *pedagogical decision rules*. We use the latter term here (*decision rules*, for short). As several developers of natural-language (NL) tutoring systems have argued, since tutorial dialogue is a form of discourse, defining effective interaction entails identifying the particular linguistic mechanisms that support learning during tutorial interaction (e.g., Boyer et al., 2010; Di Eugenio & Green, 2010; Pilkington, 2001; Ravenscroft & Pilkington, 2000). Decision rules can then be specified to guide the tutor in determining when and how to carry out these linguistic mechanisms.

This article describes the development of Rimac, a natural-language tutoring system that scaffolds students in acquiring a deeper understanding of the physics concepts and principles associated with quantitative physics problems. Rimac was designed to supplement instruction in physics tutoring systems such as Andes (e.g., VanLehn et al., 2005).[3] Rimac is primarily engineered to implement decision rules that guide the automated tutor in carrying out two linguistic mechanisms that have been found to predict learning from human tutoring: tutors' abstraction and specification of students' dialogue contributions (e.g., Katz, Allbritton, & Connelly, 2003; Ward, Connelly, Katz, Litman, & Wilson, 2009). This finding is supported by a significant body of prior research that demonstrates that the formation of abstract schema (i.e., mental representations of learned material) promotes transfer (e.g., Gick & Holyoak, 1983, 1987; Leher & Littlefield, 1993; Reed, 1993; Salomon & Perkins, 1989). During tutoring, abstraction takes place when the tutor or student relates what his or her dialogue partner said to explain a more general concept or principle. For example, during physics tutoring, abstraction involves mapping the physical state presented in a problem to concepts and principles that explain that state or to a general script for solving that type of problem. Specification is the reverse and typically occurs when the tutor (or student) distinguishes between related concepts, instantiates a formula that represents a physics principle, applies a problem-solving script to the problem at hand, and so forth.

From a linguistic perspective, abstraction and specification are often implemented through hypernym/hyponym pairs of terms (Halliday & Hasan, 1976). For example, in the following exchange from a live tutoring session, the tutor specifies "velocity" (hypernym) in the student's turn to "horizontal components of the velocity" (hyponym).

> *Example 1*
>
> *Student:* Velocity is in the same direction as acceleration so the ball is faster coming down.
>
> *Tutor:* It [the ball] slows down going up, and it speeds up coming down—but all the time the *horizontal components of the velocity* stay unchanged. [italics ours]

However, sometimes abstraction and specification are implemented through semantic relations between speaker turns, with few or no lexical cues such as those shown in Example 1, and inference is required to detect these semantic relations. For example, in the following exchange, the student needs to infer that the tutor's phrase "change in velocity" abstracts over the student's phrase "final velocity is larger than the starting velocity."

> *Example 2*
>
> *Tutor:* How do we know that we have an acceleration in this problem?
>
> *Student:* Because the final velocity is larger than the starting velocity, 0.
>
> *Tutor:* Right—a *change in velocity* implies acceleration. [italics ours]

In addition to implementing decision rules to guide the automated tutor in abstracting and specifying students' dialogue turns, Rimac also simulates a few other linguistic processes that commonly occur during physics tutoring—most notably, joint construction of conditional reasoning relations, as we will illustrate presently (Louwerse, Crossley, & Jeuniaux, 2008).

Several studies have shown that data-driven machine learning techniques such as reinforcement learning can be applied to logged interactions from natural-language tutoring systems in order to derive decision rules to guide tutorial interaction (e.g., Beck, Woolf, & Beal, 2000; M. Chi et al., 2010, 2011a, 2011b; Murray & VanLehn, 2006). Evaluations of ITSs that implement these rules have found that these systems significantly outperform counterpart systems that carry out random policies—for example, "eliciting" a problem-solving step or dialogue goal from the student sometimes, "telling" the student that step or goal at other times, without clear guidelines about what to do when. Some rule-driven tutoring systems have also outperformed systems that implement "fixed" tutoring policies (e.g., Murray & VanLehn, 2006)—for example, responding to students' help requests with increasingly directive feedback such as prompt first, then hint, then teach relevant background knowledge, and then (if all else fails) tell the student what to do (the so-called *bottom out hint*).

Although this research demonstrates the promise of automated methods for deriving effective decision rules to guide tutorial dialogue, it also shows that the process is both difficult and costly. As M. Chi et al. stated, "Finding effective tutorial tactics is not easy" (M. Chi, Jordan, VanLehn, & Litman, 2009, p. 197). In addition, the decision rules that stem from this approach are highly domain specific and difficult to interpret. Take, for example, one decision rule that M. Chi et al.'s (2011a) reinforcement-learning-based system defined for "elicit versus tell"—that is, should a tutor prompt the student for domain content at a particular point in a dialogue or tell the student that content?

> Rule 6 suggests that when the next dialogue content step is difficult (StepSimplicityPS is 0), the ratio of physics concepts to words in the tutor's turns so far is high (TuConceptsToWordsPS is 1), and the tutor has not been very wordy during the current session (TuAvgWordsSesPS is 0), then the tutor should tell. (p. 96)

On the one hand, finely nuanced rules such this one have the benefit that researchers using conventional experimental methods to test hypothesized decision rules could not predict these rules in

[3] Rimac is the name of a river whose source is in the Andes. Its name is a Quechua word meaning *talking*; hence, the nickname for Rimac, *talking river*. We thus considered the name Rimac to be well suited for a dialogue system that could be embedded within the Andes tutoring system.

the first place. Similar observations have been made of the use of automated approaches to identify linguistic features of tutorial dialogue that predict learning, such as hidden Markov models (e.g., Boyer et al., 2010). On the other hand, rules such as this are cryptic and complex to implement, as the researchers have acknowledged.

In developing natural-language dialogues for Rimac, we strove to specify decision rules that were supported by preliminary empirical research, more intuitive than those illustrated previously, and readily implementable using a common framework for generating NL dialogues, which we will describe presently. Consequently, we took a more conventional approach. We first performed correlational analyses to identify specific relations between tutors' and students' dialogue moves in a large corpus of human-tutored physics dialogues that predict student learning gains from pretest to posttest. We then examined the context in which these relations typically occur and formulated decision rules that specify these contextual conditions. We implemented these rules within Rimac and are currently evaluating the system to determine if it outperforms a less interactive, less rule-driven tutoring system control.

In the next section, we situate Rimac in a framework of tutoring research that highlights the need for effective decision rules to guide natural-language dialogue systems. In keeping with the theme of this special issue of the *Journal of Educational Psychology*, we then describe the empirical research that we conducted to derive decision rules to guide abstraction, specification, and other commonly occurring relations between students' and tutors' dialogue turns, particularly during physics tutoring, and illustrate how we implemented these rules within Rimac.

Cooperative Execution During Scaffolding

The most intensive interaction during human one-on-one tutoring takes place during *scaffolding*, which M. T. H. Chi et al. (2001) defined as follows:

> [A] scaffolding move is a kind of *guided prompting* that pushes the student a little further along the same line of thinking, rather than telling the student some new information, giving direct feedback on a student's response, or raising a new question or a new issue that is unrelated to the student's reasoning . . . . The important point to note is that scaffolding involves *cooperative execution* or *coordination* by the tutor and the student (or the adult and child) in a way that allows the student to take an increasingly larger burden in performing the skill. (p. 490).

The nexus of scaffolding lies in the fourth step of Graesser et al.'s (1995) "five-step dialogue frame" (p. 504) to describe the cyclic nature of tutorial interaction:

Step 1. Tutor asks question.

Step 2. Student answers question.

Step 3. Tutor gives short feedback on the quality of the answer.

Step 4. Tutor and student collaboratively improve the quality of the answer.

Step 5. Tutor assesses student's understanding of the answer.

As Graesser et al. (1995) and others (e.g., VanLehn et al., 2007) have noted, understanding Step 4 of this frame—that is, scaffolding to improve the student's answer—could hold the key to understanding why human tutoring is so effective.

M. T. H. Chi et al.'s (2001) definition of scaffolding names two linguistic mechanisms that drive it: *coordination* and *cooperative execution*. We consider coordination first, because more research has been devoted to describing it. *Coordination* refers to the ways in which the tutor and student "stay on the same page"—that is, "grounding" the conversation, by acknowledging their dialogue partner's moves, negotiating the meaning of terms, and sharing knowledge (Clark & Schaefer, 1989; VanLehn, 2011). Coordination can also be supported by various forms of verbal alignment, such as lexical cohesion (e.g., word repetition, synonymy, paraphrase), and syntactic (word order) alignment (Garrod & Pickering, 2004). When the student hears his words (or word order) echoed in the tutor's turn, the student knows that the tutor understood what he or she said. Several studies have shown that the degree of lexical and syntactic cohesion (alignment) during tutoring predicts learning (e.g., Litman & Forbes-Riley, 2006; Steinhauser et al., 2011; Ward & Litman, 2008, 2011), in addition to potentially enhancing coordination.

Cooperative execution refers to the joint construction of a line of reasoning. According to VanLehn (2011), cooperative execution takes place as tutors prompt students to continue a line of reasoning, indicate who should continue the execution, and accept the student's reasoning (p. 211). Our observations of tutorial dialogues reveal that cooperative execution during scaffolding involves more than these dialogue management processes; it also involves co- construction of the parts of an emerging line of reasoning or explanation. The analyses described in the Method section were motivated by our hypothesis that tutoring researchers need to formally describe these co-constructed dialogue moves and determine which types of moves support learning in order to develop natural-language dialogue systems that are as effective, or even more effective, than human tutors.

A Linguistic Framework to Describe Cooperative Execution

Rhetorical Structure Theory (RST) is a theoretical linguistic framework that specifies types of logical and functional relationships between parts of text and spoken discourse, including various types of abstraction and specification relations. Mann and Thompson (1988), who developed RST, argued that "it describes the relations among text parts in functional terms, identifying both the transition point of a relation and the extent of the items related" (p. 271). Functional and logical relationships between parts of spoken and written discourse go by many names, including *rhetorical relations*, *coherence relations*, and *discourse relations* (Hovy, 1990). We use the latter term here.

Table 1 defines and illustrates the set of abstraction/specification relations, and other discourse relations, which we manually tagged in a corpus of human tutorial dialogues in order to determine which co-constructed relations predict learning and are thereby most important to simulate in Rimac. For example, a student applies the equation for acceleration; the tutor then says something general about acceleration (e.g., "Acceleration is a vector and hence has direction as well as magnitude."). In RST, this is a jointly constructed *instance:abstract* discourse relation. To take another example, the tutor describes a set of conditions that apply to a given physical situation—for example,

Table 1
Discourse Relations Tagged in the Dialogue Corpus

Relation and definition (S = speaker)	Example
Abstraction/specification relations	
Abstract:instance (instance:abstract): S2 instantiates the abstraction stated by S1, or S2 abstracts over the information presented by S1.	*Tutor:* How can the acceleration be 0 if there are forces on it? *Student:* The sum of the forces equal 0 for there to be no acceleration. *Tutor:* That's exactly right. The weight and the normal force are (in this case) equal and opposite. Explanation: "In this case" (as the tutor says), the weight and normal force being equal and opposite represent an instance of the abstraction "sum of forces equal 0."
Set:member (member:set): S2 presents a member of the set referred to by S1, or S2 names the set to which an item mentioned by S1 belongs.	*Tutor:* What does the problem ask for? *Student:* The magnitude of the acceleration *Tutor:* What type of acceleration? *Student:* Average Explanation: The tutor refers to acceleration as a set and prompts for a member of that set; the student gives the type of acceleration asked for in the problem.
Whole:part (part:whole): S2 names a part of an object that S1referred to, or S1 names a part of an object named by S2. (In physics, "parts" are often vector components or the specific forces acting on an object.)	*Student:* Acceleration would be plus. *Tutor:* Right, the x component of the acceleration would be plus. Explanation: The student names a vector (acceleration); the tutor refers to a specific component of that vector.
Process:step (step:process): S2 presents a step that follows from the process or line of reasoning described by S1, or S2 describes the line of reasoning that leads to the step described by S1.	*Student:* The acceleration is 0. *Tutor:* So then $m^{*}a = 0 = F_{net} = T - W$ and hence $T = W$. Explanation: The student gives a step in a line of reasoning; the tutor expands the line of reasoning (process) that follows from that step.
Object:attribute (units, direction, magnitude): S1 names an object or value; S2 specifies a property of that object—in particular, its units, direction, or magnitude.	*Student:* Velocity is 14. *Tutor:* Right, 14 m/s. Explanation: The student provides a value for velocity; the tutor specifies its units.
Term:definition (definition:term): S2 defines a term mentioned by S1, or S2 labels a statement by S1 with an appropriate term.	*Tutor:* What is the definition of the average acceleration (in words or in mathematics)? *Student:* $A = (Vf - Vo)/Tf - To$. Explanation: The tutor prompts the student to define average acceleration; the student does so.
General:specific (specific:general): S2 names a state, object, or action that is related to the content in S1 but is more specific, or S2 is more general than the state, object, or action referred to in S1. Applies when none of the preceding relations apply.	*Student:* Average acceleration can vary. *Tutor:* Right; it can go up above the average and down below it. Explanation: The tutor specifies how acceleration can vary.
Other commonly occurring relations in physics tutoring	
Condition:situation (situation:condition): (a) S1 presents a condition or set of circumstances, and S2 states the situation that stems from or coincides with those conditions, or (b) S1 presents a situation, and S2 states the conditions or circumstances that explain that situation.	*Tutor:* When do kinematics equations apply? *Student:* When the acceleration is constant. Explanation: This relation could be stated in conditional form: if acceleration is constant, then the kinematics equations apply.
Compare: S2 compares an object, situation, or value referred to by S1 with some other object, situation, or value.	*Tutor:* What is the net force that the air bag imparts to the driver? *Student:* Equal to the force the driver applies to the airbag. *Tutor:* Same direction? *Student:* No, opposite direction. Explanation: The tutor prompts the student to compare the value and direction of two.

"A car is moving to the right and is suddenly stopped"—and then prompts the student to state the situation that follows from this set of conditions—for example, that the car's acceleration is to the left. This is a co-constructed *condition:situation* (conditional) relation. Any relation can be delivered didactically, by the tutor or student, instead of interactively, as in these examples. For example, the tutor could have stated the same conditional relation didactically as follows: "Since the car is moving to the right and is suddenly stopped, its acceleration is to the left." However, we focused our investigation on the potential relationship between co-constructed discourse relations and learning because these relations realize cooperative execution during scaffolding.

Method

To reiterate, our goals in the analyses described in this section were to (a) determine if the frequency of particular types of co-constructed discourse relations (those described and illustrated in Table 1) predict learning, and whether this varies by student ability level, and (b) formulate decision rules that specify the context in which those

discourse relations predicting learning occurs, so that these rules can guide student–tutor interaction in a NL tutoring system (Rimac). Toward these aims, we coded all instances of co-constructed discourse relations in a large corpus of human-tutored physics dialogues. The dialogue corpus and our approach to coding identified relations are described in this section.

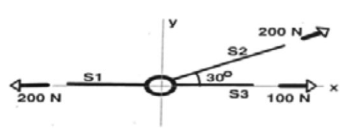
Dialogue Corpus

A well-known problem in physics education is that many students learn to apply scripts for solving particular types of problems and succeed in college-level physics courses; however, they nonetheless leave these courses without understanding fundamental physics concepts and principles (Halloun & Hestenes, 1985). Reflective discussions following problem-solving exercises encourage students to think about the concepts and principles associated with quantitative problems, often by changing some aspect of the problem and prompting the student to consider how the answer would change, as illustrated in Table 2. Several studies have demonstrated the instructional benefits of reflection on problem-solving exercises (e.g., Collins & Brown, 1986; Katz, Connelly, & Wilson, 2007; Katz et al., 2003; Lee & Hutchison, 1998; Tchetagni, Nkambou, & Bourdeau, 2007; Ward & Litman, 2011).

The dialogue corpus that we analyzed stems from previous research in which we compared the effectiveness of human-guided reflective discussions about physics problems solved within the Andes physics tutoring system (VanLehn et al., 2005) with static text explanations and a no-dialogue control. We summarize the data collection procedures that produced the dialogue corpus in this section. More details about the study can be found in Katz et al. (2003).

Table 2
Example of a Reflective Dialogue Between a Human Tutor and Student

Problem: In the figure below, each of the three strings exerts a tension force on the ring as marked. Use the labels S1, S2, and S3 to refer to the three strings. Find the components of the net force acting on the ring.



Reflection question: What if I now told you that this ring has an acceleration. If you knew the mass of the ring (3 kg), how would you solve for the acceleration?

Student: 73.2 − 3*a; 100 − Fw = 3*a. Is this right; how would the acceleration be the same for both?

Tutor: You have to keep the *a_x* and *a_y* distinguished. They are two completely independent numbers that (together with *a_z*) specify your acceleration vector. You don't try to boil them down to one number. It's as if I told you, "To get to my house, you go 3 blocks north and 5 blocks east," and you said, "Ah, so you just go 8 blocks"—the two numbers together are the vector; they don't "boil down" to one number. OK?

Student: But can't it only have one acceleration?

Tutor: It does have only one acceleration, but that acceleration is a vector and it takes 3 numbers to write it down. You need to review vectors in some detail; *a_x, a_y,* and a_z together specify the acceleration vector.

Note. This example problem is part of the Andes Physics Tutor system, which was developed at Arizona State University and the University of Pittsburgh with support from the Pittsburgh Science of Learning Center, National Science Foundation Award SBE-0836012, and Office of Naval Research Grant N00014-96-1-0260 and is available at http://www.andestutor.org

Students who were taking an introductory physics course at the University of Pittsburgh first took a physics pretest, with nine quantitative and 27 qualitative physics problems. Following the pretest, students reviewed a workbook chapter developed for the experiment and then received training on using Andes. There were three conditions: one in which students received reflection questions and interacted with a human tutor via a chat interface; a second reflection condition in which students were asked the same set of reflection questions but received a static text explanation as feedback after they responded to these questions; and a third, a control condition in which students were not asked reflection questions but solved more problems than students in the other two conditions to control for time on task. There were 15 students in the static text and control conditions and 16 students in the human-tutored condition. In the correlational analyses discussed here, we only analyzed data from the human-tutored condition, since we were interested in modeling effective aspects of human tutorial dialogue.

Students in each condition began by solving a problem in Andes. After completing the problem, students in both the static feedback and human-tutored conditions were presented with a conceptually oriented reflection question, as illustrated in Table 2. Reflection questions such as the one shown in Table 2 are not part of Andes; they were added for the experiment. After a student in the human-tutored condition entered a response to the reflection question, the student engaged in a typed dialogue with his or her tutor via a simple chat interface. This dialogue continued until the tutor was satisfied that the student understood the correct answer to the question.

Between three and eight reflection questions were asked per problem solved in Andes for a total of 12 problems. After completing these problems and their corresponding reflective dialogues, students took a posttest that was isomorphic to the pretest, and the test order was counterbalanced. The main finding of the study was that students who answered reflection questions learned more than students in the no-reflection control, who solved more Andes problems (Katz et al., 2003). Consistent with authors of several other studies who found a null effect for the interaction hypothesis, we did not observe a significant difference between the static feedback and human-tutored conditions (VanLehn, 2011; VanLehn et al., 2007). However, the human-tutored dialogue corpus revealed abundant instances of highly interactive, cooperative execution during scaffolding episodes—specifically, exchanges in which the tutor incorporated parts of the student's turn, built on the student's turn, and so on (e.g., Table 2)—or less frequently, the student did the same with respect to a preceding tutor turn. Hence, we deemed this corpus well-suited for exploring correlations between interactivity and student learning outcomes.

The dialogue corpus is sizeable. Among the 16 students in the human-tutored condition (four men, 12 women), 15 completed all 60 reflection question dialogues with a human tutor; one student participated in 53 dialogues, producing a total of 953 reflective dialogues. There were a total of 2,218 student turns and 2,135 tutor turns across dialogues. The average number of turns per reflective dialogue was 4.6, ranging from 2.1 turns for simple reflection questions to 11.4 turns for the most complex questions. All dia-

logue examples presented in this article stem from this tutoring corpus, unless otherwise noted.

Coding Scheme

Within each reflective dialogue, all student and tutor turns were first manually parsed into clauses. We then searched for co-constructed discourse relations at the exchange level—that is, between a tutor's dialogue turn and the subsequent student turn, or the reverse. We coded these relations at two levels of analysis: abstraction level type, and discourse relation type.

Abstraction Level Type

At the coarsest level, we tagged the level of abstraction of each exchange in which a discourse relation was co-constructed. Four codes distinguish these levels of abstraction, as described in the following. Code abbreviations are shown in parentheses.

Specific-to-general (spec:gen). This code refers to abstraction, which happens in two main ways. The first type is when the second speaker refers to a more general concept, principle, or value than one that the first speaker referenced in his dialogue turn. For example, in the following exchange, the tutor refers to speed, and the student classifies speed as a scalar quantity:

Example 3

Tutor: Since the question asked about SPEED, suppose we had found *v_y* to be negative. Should we include the minus sign when giving the speed?

Student: I would say no because speed is scalar and doesn't include direction.

In the second type of abstraction, the second speaker refers to a physics principle that explains or is illustrated by problem-specific content in the first speaker's turn. For example, in the following exchange, the tutor prompts the student to apply a principle about the relationship between acceleration and velocity to the bullet in the case at hand:

Example 4

Reflection question: The bullet is travelling to the right. What direction is its acceleration?

Student: To the left because it is making the bullet slow down.

Tutor: Good—when something is slowing down, its acceleration has a component opposite to its velocity.

General-to-specific (gen:spec). This code refers to specification, which is the inverse of abstraction and also happens in two main ways. The first type is when the second speaker refers to a more specific concept, principle, or value than the one to which the first speaker referred. For example, in the following exchange, the tutor asks for the forces on a climber, and the student names two types of forces:

Example 5

Tutor: What are the *forces* on her?

Student: Her *weight* and the *tension* of the rope. [italics ours]

The second main type of specification is when the second speaker instantiates a principle or concept to which the first speaker refers. For example, in the following exchange, the student carries out the tutor's directive to apply Newton's second law to the current problem:

Example 6

Tutor: Now use Newton's second law and find [the climber's] acceleration—a number and units; show me the symbols (the algebra).

Student: 39/55 = *a*, *a* = .71 m/s^2 downward.

Specific (spec). This code refers to cases in which the student and tutor are both speaking at the same level of abstraction, typically in reference to a particular problem. For example, in the following exchange, the tutor and the student refer to the bungee in the current problem. The tutor explains the situation that would result from the student's erroneous claim via a co-constructed conditional relation:

Example 7

Student: The only force acting on the bungee is the weight of the person.

Tutor: If that were true, the bungee would accelerate downward!

General (gen). This code refers to cases in which the student and tutor both speak at an abstract level, referring to principles, laws, definitions, and so forth that are not directly tied to a particular problem. For example, in the following exchange, the tutor and student step outside of the context of the current problem (about a falling hailstone) to discuss the difference between distance and displacement, in this comparison relation:

Example 8

Tutor: Is there a difference between displacement and distance?

Student: The displacement can have either value [+ or −], but distance is only +.

Table 3 presents the mean and standard deviation of abstraction level tags across subjects.

Discourse Relation Type

At a finer level of analysis, we tagged the dialogue corpus for the particular types of abstraction and specification relations defined and illustrated in Table 1, in addition to two other commonly occurring discourse relations in physics tutoring dialogues—conditional reasoning statements and comparisons. Most of these discourse relations are bidirectional (e.g., set:member, member:

Table 3
*Mean Frequency of Abstraction Level Tags Across Tutored Subjects (*N *= 16)*

Abstraction level	Mean	*SD*
Specific-to-general	14.13	4.83
General-to-specific	37.31	15.12
Specific	3.31	2.18
General	11.06	5.31

set); the exceptions are object:attribute and compare. We tagged bidirectional relations separately (e.g., we treated set:member and member:set as individual relations) and also treated each of the object:attribute categories as a separate relation. Hence, overall, there are 17 discourse relations in our coding scheme.

The basic unit of analysis at the discourse relation level is one of these codes, specified in two ways. First, we specify the *direction* of the co-constructed relation in the exchange—that is, does the tutor (T) start the relation and then the student (S) completes it, or the reverse? The former is indicated by T–S before the discourse relation name, and the latter by S–T—for example, S–T set: member represents a set:member relation that the student initiates and the tutor completes; and T–S abstract:instance represents an abstract:instance relation that the tutor initiates and the student completes. To illustrate, in the example shown in Table 1 for set:member, the second exchange (T: What type of acceleration? S: Average) would be tagged as T–S set:member.

The second way in which we modify discourse relation tags is by indicating whether the second turn in a tagged relation was prompted, via a question, or initiated by the second speaker. Prompted relations, such as the one for set:member, are unmodified—that is, T–S set: member means that the tutor prompted the student to provide a member of a named set, as in the preceding example about "type of acceleration." Initiated relations are flagged as elaborations (elab), because the second speaker is adding information to what the first speaker said. To illustrate, in the example for abstract:instance shown in Table 1, the tutor elaborates on the student's turn, by instantiating the student's abstract statement:

> *Example 9*
>
> *Student:* The sum of the forces equals 0 for there to be no acceleration.
>
> *Tutor:* That's exactly right. The weight and the normal force are (in this case) equal and opposite.

This relation would be tagged as S–T elab(abstract:instance) to indicate that the tutor elaborated on the student's statement via an abstract:instance relation. Instantiation is signaled by the tutor's phrase "in this case."

In addition to prompted and initiated variants of discourse relations, in both directions (S–T and T–S), we included three types of aggregate variables in our analyses. One aggregate variable includes the four prompted and initiated (elaborated) forms of a discourse relation. For example, the aggregate variable *whole:part* represents:

> S–T whole:part + T–S whole:part + S–T elab(whole:part) + T–S elab(whole:part).

The second type of aggregate variable includes the four forms of the first relation, plus the four forms of its inverse. For example, the following formula represents *all-whole:part-bd*, where *bd* means bidirectional, for a particular relation (e.g., whole:part and part:whole, each consisting of the four forms shown in the formula):

> [S–T whole:part + T–S whole:part + S–T elab(whole:part) + T–S elab(whole:part)] + [S–T part:whole + T–S part:whole + S–T elab (part:whole) + T–S elab(part:whole)].

The third type of aggregate variable includes the summation of all initiated elaborations. Specifically, T-S elab is the summation of student elaborations on the tutor's previous turn, for all base relations (e.g., whole:part, set:member); S-T elab is the summation of tutor elaborations of the student's previous turn, for all base relations; and all-elab-bd = T-S elab + S-T elab.

Table 4 summarizes the means and standard deviation of discourse relation tags and aggregate tags across subjects.

Data Analysis

We conducted correlational analyses between the frequency of abstraction level codes, discourse relation codes, and three measures of student learning: overall gain score from pretest to posttest, gain score on qualitative test items, and gain score on quantitative test items. We conducted these analyses taking the 16 tutored students as a whole and separately for low and high pretest students, as classified according to a median split. There were seven high pretest students and nine low pretest students. These numbers are uneven because the two pretest scores in the middle of the distribution were identical; both students who had these scores were assigned to the low pretest group. We divided students into these ability groups in order to investigate whether better prepared students (high pretesters) might benefit from co-constructing dif-

Table 4
Mean Frequency of Discourse Relation Tags Across Tutored Subjects (N = 16)

Discourse relation variable or aggregate variable	Mean	*SD*
Abstract:instance	9.63	5.28
Instance:abstract	3.50	2.34
All-abstract:instance-bd	13.13	6.02
All-compare	3.19	1.83
Term:definition	3.00	2.25
Definition:term	0.13	0.34
All-term:definition-bd	3.13	2.22
Object:attribute-units	1.63	2.06
Object:attribute-direction	4.06	2.41
Object:attribute-sign	0.19	0.40
Object:attribute-magnitude	0.69	0.79
All-object-attribute	6.56	3.76
Process:step	0.56	0.63
Step:process	3.00	2.34
All-process:step-bd	3.56	2.31
Set:member	0.88	1.50
Member:set	2.00	1.67
All-member:set-bd	2.88	2.58
Whole:part	2.88	1.78
Part:whole	0.44	0.73
All-part:whole-bd	3.31	1.99
Circumstance:situation	13.00	6.79
Situation:circumstance	9.19	2.90
All-circumstance:situation-bd	22.19	6.93
Gen:spec	3.31	1.99
Spec:gen	0.75	1.07
All-gen:spec-bd	4.06	2.24
T–S elab	1.00	1.10
S–T elab	22.13	12.15
All-elab-bd	23.13	12.41

Note. Aggregate variables include modified forms of the base relation (e.g., whole:part) as described in the text. Gen = general; Spec = specific; T = tutor; S = student; bd = bidirectional; T-S elab = summation of student elaborations on the tutor's previous turn, for all base relations; S-T elab = summation of tutor elaborations of the student's previous turn, for all base relations; all-elab-bd = T-S elab + S-T elab.

ferent types of discourse relations with their tutor than less well-prepared students (low pretesters).

The results of these analyses are presented in the next section. We then describe the decision rules that stem from these findings.

Results and Discussion

Discourse Relations That Predict Learning: All Students Considered Together

Correlations for the subject pool taken as a whole ($N = 16$) are displayed in Table 5. To save space, we only discuss significant findings ($p \leq .05$) for all three types of gain.

Overall gain. The frequency of three discourse relations predicted overall gain: (a) various forms of the whole:part relation [S–T elab(whole:part) and two aggregate variables: whole:part and all-whole:part-bd], (b) S–T situation:condition relations, in which the student prompts the tutor to specify the conditions under which a physical situation occurs and the tutor replies accordingly, and (c) various forms of the step:process relation [S–T elab(step: process) and the aggregate variable step:process], in which one dialogue partner provides the steps in a line of reasoning that stem from, or lead to, a step in his partner's turn, for example:

Example 10

Reflection question: How do we know that we have an acceleration in this problem?

Student: Because of gravity pulling down.

Tutor: The force due to gravity produces a net force and thus an acceleration.

In this exchange, the tutor provides the line of reasoning that follows from the student's response (gravity→ existence of a net force → existence of acceleration), via an S–T elab(step:process) relation.

Qualitative gain. Generalizations predicted learning of a qualitative (conceptual) nature; a trend was also found for generalization and overall gain. This is not surprising, given that generalizations typically address physics concepts, laws, and principles. As with overall gain, various forms of the step:process relation also predicted qualitative gain across subjects [S–T elab-(step:process) and two aggregate variables: step:process and all-process-step-bd]. In addition, a particular type of generalization predicted qualitative gain: S–T elab(member:set), in which the tutor elaborates on a student turn by stating the set to which an object that the student referred to belongs:

Example 11

Reflection question: How do we know that we have an acceleration in this problem?

Student: Because it is a free fall problem so gravity is at work.

Tutor: Gravity is a type of acceleration.

Quantitative gain. The "spec" abstraction level type, representing exchanges in which the tutor and student refer to the current problem, negatively correlated with quantitative gain. However, two particular forms of specification strongly predicted quantitative gain: S–T elab(set:member), in which the tutor states a member of a set that the student referred to, and S–T elab(whole: part), which typically reflects exchanges in which the tutor specifies the components of a vector that the student mentioned or the applied forces on an object that the student mentioned:

Example 12

Student: (String1 + String2)/g = mass of plane.

Tutor: It would be $(F_{t1\_y} + F_{t2\_y})/g$ = mass, OK?

Table 5
Correlations for All Students Considered Together (N = 16)

Abstraction level and discourse relations	Mean	*SD*	*R*	*p*
Overall gain				
Abstraction level: [spec:gen]	14.13	4.829	.450	.081
Discourse relations				
S–T elab(step:process)	1.56	1.365	.646	.007**
step:process	3.00	2.338	.582	.018
S–T elab(member:set)	0.94	0.680	.667	.005**
S–T elab(whole:part)	1.00	1.366	.524	.037
whole:part	2.88	1.784	.528	.035
all-part:whole-bd	3.31	1.991	.553	.026
S–T situation:condition	0.44	0.814	.531	.034
[definition:term]	0.13	0.342	−.485	.057
[all-proc:step-bd]	3.56	2.308	.473	.064
Qualitative gain				
Abstraction level: spec:gen	14.13	4.829	.516	.041
Discourse relations				
S–T elab(step:process)	1.56	1.365	.653	.006**
step:process	3.00	2.338	.591	.016
all-proc:step-bd	3.56	2.308	.527	.036
S–T elab(member:set)	0.94	0.680	.558	.025
[T–S elab(term:definition)]	0.06	0.250	.469	.067
[definition:term]	0.13	0.342	−.443	.086
[S–T step:process]	0.06	0.250	.469	.067
[whole:part]	2.88	1.784	.463	.071
[all-part:whole-bd]	3.31	1.991	.457	.075
[S–T situation:condition]	0.44	0.814	.487	.056
Quantitative gain				
Abstraction level: spec	3.31	2.182	−.530	.035
Discourse relations				
S–T elab(set:member)	0.06	0.250	.740	.001**
S–T elab(whole:part)	1.00	1.366	.675	.004**
[T–S instance:abstract]	0.38	0.500	−.493	.052
[Object:attribute-magnitude]	0.69	0.793	−.467	.068
[Process:step]	0.56	0.629	−.445	.084
[S–T elab(member:set)]	0.94	0.680	.443	.086
[all-part:whole-bd]	3.31	1.991	.452	.079

Note. Trends are indicated by brackets. Gen = general; Spec = specific; T = tutor; S = student; bd = bidirectional; elab = elaborated; proc = process.
** $p < .01$.

Discourse Relations That Predict Learning Among Low Pretest Students

Correlations for low pretest students ($N = 9$) are displayed in Table 6. We again focus our discussion on significant findings ($p \leq .05$) for all three types of gain.

Overall gain. Student generalizations over the tutor's turn positively correlated with low pretesters' overall gain score; how-

Table 6
Correlations for Low-Pretest Students (N = 9)

Abstraction level and discourse relations	Mean	*SD*	*R*	*p*
Overall gain				
Abstraction level				
T–S spec:gen	4.44	3.005	.671	.048
S–T spec	1.67	1.732	−.719	.029
Discourse relations				
S–T situation:condition	0.67	1.000	.679	.044
[S–T elab(abstract:instance)]	1.89	1.833	−.624	.072
[T–S member:set]	0.78	1.093	.617	.077
[S–T elab(member:set)]	1.11	0.782	.646	.060
Qualitative gain				
Abstraction level				
T–S spec:gen	4.44	3.005	.855	.003**
[spec:gen]	16.00	5.050	.600	.088
Discourse relations				
situation:condition	8.89	2.667	.676	.045
[S–T elab(abstract:instance)]	1.89	1.833	−.661	.053
[S–T elab(step:process)]	2.22	1.481	.594	.092
Quantitative gain				
Abstraction level				
spec	3.33	2.550	−.699	.036
[T–S gen:spec]	31.33	11.424	−.662	.052
Discourse relations				
T–S object:attribute-direction	3.33	2.121	−.672	.047
S–T elab(set:member)	0.11	0.333	.884	.002**
S–T elab(whole:part)	1.44	1.590	.741	.022
S–T elab(gen:spec)	1.78	2.279	.680	.044
[object:attribute-direction]	4.56	2.789	−.595	.091
[situation:condition]	8.89	2.667	−.657	.055

Note. Trends are indicated by brackets. T = tutor; S = student; gen = general; spec = specific; elab = elaboration.
** $p < .01$.

ever, tutor specifications relative to the tutor's turn negatively correlated with overall gain. Consistent with the findings from the set of students taken together, one discourse relation whose frequency predicted overall gain among low pretesters was S–T situation:condition, in which the student asks the tutor to explain the circumstances under which a given physical state (velocity decreasing in the *y* direction) applies:

Example 13

Student: Why is velocity decreasing in the *y* direction?

Tutor: It starts out going up and gravity pulls it down. When acceleration is opposed to velocity, the object slows down.

Qualitative gain. Low pretesters' abstraction over the tutors' turns (T–S spec:gen) predicted qualitative gain score, consistent with a trend for abstraction either by the student or the tutor (spec:gen) to predict qualitative gain. Only one aggregate discourse relation variable significantly predicted qualitative gain among low pretest students: situation:condition, which is the conditional relation in which the second speaker provides the conditions that explain the situation described by the first speaker, either because the first speaker solicited this information or the second speaker initiated it. Example 13 illustrated a student-solicited conditional relation. The following exchange shows a tutor prompting the student to specify a condition in a T–S situation: condition relation:

Example 14

Tutor: Why does the tension equal the weight in this problem?

Student: Because there are no other outside forces acting on the bungee/jumper system.

Encouraging low pretest students to explain their claims (e.g., tension = weight) appears to be beneficial and is under the tutoring system's control, in contrast to student-initiated conditionals, such as the one shown in Example 13.

Quantitative gain. Consistent with the findings for all students considered together, the frequency of exchanges in which both participants focused on the case at hand negatively correlated with quantitative gain among low pretest students. In addition, the frequency of one type of specification negatively predicted quantitative gain for this group: T–S object:attribute-direction relations, in which the tutor prompts the student to specify the direction of a value. Specifying the correct direction of a vector often requires conceptual understanding, so this negative correlation could reflect the difficulty that less-prepared students have in determining direction. However, the frequency of several other specification relations predicted quantitative gains for low pretesters—in particular, tutor-initiated set:member [S–T elab(set:member)], whole: part [S–T elab(whole:part)], and gen:spec [S–T elab(gen:spec)] relations. The following exchange illustrates the tutor adding more specific information to the student's dialogue turn, in an S–T elab(gen:spec) relation:

Example 15

Reflection question: Does gravity have any effect on the vertical motion of the firecracker? What about the horizontal motion? Explain your answers.

Student: Vertical motion, yes; it makes it harder for the firecracker to travel away from the earth because gravity is pushing down, so it adds resistance.

Tutor: Good, that is right (and it pulls the firecracker back down after the high point also).

As this exchange illustrates, students sometimes answer questions correctly but not completely. The tutor added information necessary to complete the student's answer to the reflection question. Perhaps making low pretest students aware of complete answers, by adding to students' dialogue contributions, increases these students' quantitative problem-solving ability.

Discourse Relations That Predict Learning Among High Pretest Students

Correlations for high pretest students ($N = 7$) are displayed in Table 7. We again focus our discussion on significant findings ($p \leq .05$) for all three types of gain.

Overall gain. The frequency of only one discourse relation significantly predicted high pretest students' overall gain score: S–T elab(whole:part), which was also observed for the group of students as a whole. As discussed previously, this relation typically occurs when the tutor specifies the components of a vector that the student named, the specific forces that comprise the net force, etc. This finding suggests that adding this level of precision to high pretesters' dialogue contributions supports learning.

Table 7
Correlations for High Pretest Students (N = 7)

Abstraction level and discourse relations	Mean	*SD*	*R*	*p*
Overall gain				
Abstraction level: [S–T Gen]	3.00	1.826	−.700	.080
Discourse relations				
S-T elab(whole:part)	0.43	0.787	.826	.022
[object:attribute-units]	1.14	0.690	.709	.074
[object:attribute-direction]	3.43	1.813	−.713	.072
[all-object-attribute]	5.43	1.272	−.695	.083
[S–T elab(member:set)]	0.71	0.488	.719	.069
Qualitative gain				
Abstraction level				
[T–S spec]	2.29	1.704	.723	.066
[T–S spec:gen]	4.86	2.734	−.733	.061
[Gen]	10.43	5.224	−.688	.087
Discourse relations				
S–T elab(term:definition)	0.29	0.756	.863	.012
object:attribute-units	1.14	0.690	.817	.025
S–T whole:part	0.14	0.378	.863	.012
S–T elab(whole:part)	0.43	0.787	.809	.028
T–S elab(condition:situation)	0.29	0.756	.863	.012
[situation:condition]	9.57	3.359	−.697	.082
Quantitative gain				
Abstraction level: [Gen:spec]	32.00	11.986	−.720	.068
Discourse relations				
T–S step:process	1.00	1.000	.831	.020
S–T elab	18.00	6.325	−.756	.049
all-elab-bd	19.00	6.952	−.762	.046
[S–T elab(instance:abstract)]	2.57	1.397	−.733	.061
[all-abstract:instance-bd]	11.71	5.707	−.739	.058
[step:process]	1.71	1.113	.694	.084

Note. Trends are indicated by brackets. S = student; T = tutor; Gen = general; elab = elaboration; bd = bidirectional.

Qualitative gain. The frequency of several discourse relations predicted qualitative gains among high pretest students: tutor definitions of terms mentioned in the student's dialogue move [S–T elab(term:definition)]; whole:part relations [S–T whole:part, and S–T elab(whole:part)]; conditional relations that the student takes the initiative to complete [T–S elab(condition:situation)]; and one aggregate variable—object:attribute-units, in which the tutor prompts the student to provide missing units, or does this for the student. Tutor-initiated definitions typically occurred when the student used a term incorrectly and the tutor corrected it, as illustrated in the following exchange:

Example 16

Student: The force equals the mass of the book plus the other forces acting on it, which would be considered the acceleration.

Tutor: Well . . . the acceleration is the rate of change of its velocity.

Perhaps giving high pretest students the definition of a misused term sometimes suffices to correct their knowledge.

It is unclear why providing units (or prompting students to provide units) might support qualitative understanding. Perhaps units cement the difference between concepts or support students in understanding the temporal and spatial properties of physical concepts.

Quantitative gain. The frequency of one discourse relation predicted quantitative learning among high pretest students: exchanges in which the tutor provides a step in a line of reasoning and prompts the student to provide the line of reasoning that follows from that step or that is necessary to get to that step. For example, in the following exchange, the tutor states the final step in the problem (tension = weight) and prompts the student to explain how she arrived at that conclusion, via a T–S step:process relation:

Example 17

Tutor: OK . . . so then why does tension = weight . . . show me how you got your answer.

Student: $F = F_{\text{ten}} - ma$, $a = 0$, so $mg = F_{\text{ten}}$.

Two aggregate variables indicate that elaborations potentially hinder high pretesters' ability to gain quantitative knowledge and skills: S − T elab and all-elab-bd. The latter includes all elaborations initiated by either students or tutors; however, most were issued by tutors (354 vs. 16). Perhaps filling in too many details in the line of reasoning hinders learning among more knowledgeable students; it might be better to let them fill in the gaps on their own, as indicated by prior research on textual coherence (e.g., McNamara, 2001; McNamara & Kintsch, 1996; McNamara, Kintsch, Songer, & Kintsch, 1996).

Decision Rules to Guide Automated Scaffolding

The analyses discussed in the previous section suggest that particular forms of cooperative execution that take place during scaffolding, implemented via co-constructed discourse relations, predict learning gains. However, since correlation does not imply causality, we need to determine if a tutorial dialogue system that is explicitly designed to encourage joint construction of these potentially beneficial discourse relations outperforms a counterpart tutoring system not so designed.

This section describes decision rules that stem from the findings discussed in the preceding section. These rules can guide the tutoring system in simulating these potentially effective aspects of human tutoring. Where appropriate, we provide further detail on the context in which these rules apply than we did in the previous section. In the next section, we illustrate how these decision rules are implemented in Rimac, in contrast to a control dialogue system.

Rule 1. *When the student provides a step in a line of reasoning, the tutor may provide the missing steps of the line of reasoning, rather than ask about each step individually.*

This decision rule stems from several correlations involving the step:process relation—specifically, for the group of students taken as a whole, the frequency of S–T elab(step:process) relations predicted overall gain, $R(14) = .646$, $p = .007$, and the aggregate variable step:process predicted both overall gain and qualitative gain, $R(14) = .582$, $p = .18$, and $R(14) = .591$, $p = .016$, respectively. The tutor's extension of the student's line of reasoning took place in three main contexts: (a) when the student answered a question correctly but not completely, as illustrated in Example 10; (b) when the student had some trouble coming up with a problem-solving or reasoning step, in which case the tutor filled in some of the line of reasoning and then prompted the student for additional steps; and (c) when the student reached the final step of a solution or line of reasoning,

in which case the tutor summarized the steps leading up to that conclusion. This mainly happened at the end of a problem.

Rule 2. *If a student states a value but does not state how he derived it, the tutor should prompt the student to explicate his reasoning process.*

This rule is similar to the preceding one, except that here the student, not the tutor, is expanding the line of reasoning as illustrated in Example 17. It stems from the finding that the frequency of T–S step:process relations predicted quantitative learning gains, particularly for high pretest students, $R(5) = .831$, $p = .020$.

Rule 3. *When students state vectors rather than vector components while solving equations, the tutor should provide the corresponding equation with components. Alternatively, the tutor should prompt the student to provide the vector components.*

This rule stems from several correlations involving the basic whole:part relation. For example, the frequency of S–T elab(whole:part) relations, in which the tutor specifies the vector components (Example 12), predicted overall gain for the whole group of students, $R(14) = .524$, $p = .037$. In addition, two aggregate variables predicted overall gain: whole:part and all-whole:part-bd, $R(14) = .528$, $p = .035$, and $R(14) = .553$, $p = .026$, respectively. Similar correlations were found for the group of high pretest students.

Rule 4. *When the student oversimplifies the circumstances under which a given physical situation applies or fails to make explicit the relationship between a narrower term and a broader term, the tutor should make these "member:set" relations explicit.*

This rule is based on the finding that the frequency of S–T elab(member:set) relations predicted overall gain for all students taken together, $R(14) = .667$, $p = .005$, for low pretest students, $R(7) = .646$, $p = .060$, and for high pretest students, $R(5) = 0.719$, $p = .069$. Example 11 illustrates a case in which the tutor states the class in which a narrower concept belongs (e.g., gravity is a type of acceleration) when the student's claim implies this but does not say it explicitly.

The following exchange illustrates the tutor reacting to a student's oversimplification of the circumstances associated with a physical situation. The student provides two examples of forces that could account for constant velocity (or a null net force); the tutor names the set "Anything else [other forces] that could make the net force 0":

Example 18

Student: No acceleration for a constant velocity; this would only be possible for a situation with a great deal of air resistance or friction.

Tutor: Or anything else to make the net force 0! The forces could be different.

Rule 5. *The tutor should ask "why" questions when the student does not provide an explanation to support a claim, especially with less knowledgeable students.*

This rule stems mainly from our finding that the frequency of conditional relations in which the tutor specified the conditions under which a situation described by the student applied (i.e., S–T situation:condition relations), correlated with overall learning gains for the group of low pretest students, $R(7) = .679$, $p = .044$. The aggregate variable situation:condition also predicted qualitative gains for this group, $R(7) = .676$, $p = .045$. This finding supports Louwerse et al.'s (2008) suggestion that prompting students to express conditional relations exposes gaps in their reasoning process that the tutor can address, and this exercise promotes learning.

Example 13 illustrates a case in which a student takes initiative and asks the tutor to state the conditions that explain a given situation, while Example 14 illustrates the more readily implemented case of the tutor prompting the student to state relevant conditions, via a T–S situation:condition relation. "Why" prompts such as this typically occur when the student answers a question correctly but does not justify his answer, as in the following exchange:

Example 19

Reflection question: Does average acceleration imply that the acceleration is the same at every instant?

Student: No.

Tutor: Correct—could you say why?

Student: Because average is taking different velocities over different times.

Rule 6. *If the student answers a question incorrectly, if possible show why it is incorrect by stating the conditions under which it would be correct.*

This rule is related to the preceding and is mainly motivated by the correlation between the frequency of the aggregate situation: circumstance relation and qualitative gains among low pretest students. It reflects cases in which a student states a situation (the consequent in a conditional relation) and the tutor provides the conditions (antecedent) that would hold true if the situation were true. For example, in the following dialogue excerpt, the tutor states the conditions that would explain a net force of 0 on a bungee jumper:

Example 20

Reflection question: What minimum acceleration (in magnitude) must the jumper have in order for the cord not to break while he is on his way down?

Student: 700 N/mass = a.

Tutor: Not quite, good start. What is the *net* force on him? (in terms of the tension and mg)?

Student: The net force is 0.

Tutor: Ah, OK. When he is hanging there, it is 0, or if he is moving with constant velocity.

Rule 7. *If the student gives a partially correct answer, the tutor should complete it, especially for less knowledgeable students.*

This rule is based on the finding that the frequency in which the tutor extends a partial or underspecified statement in the student's dialogue turn, via S–T elab(gen:spec) relations, correlated with quantitative gains, among low pretest students. Example 15 demonstrates a tutor's application of this rule.

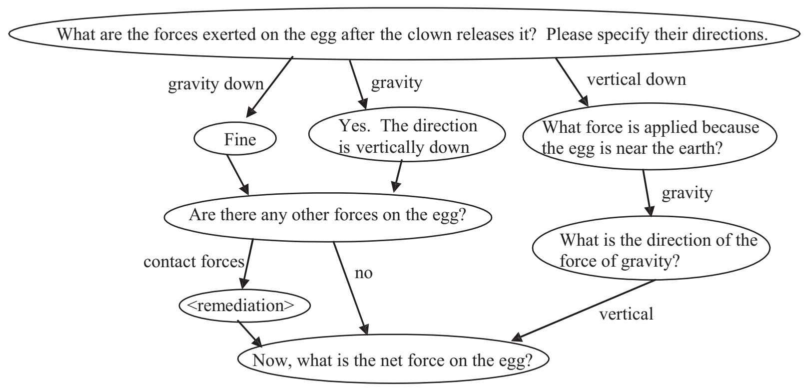


Figure 1. The dialogue paths of three students as they traverse the arcs in a knowledge construction dialogue (KCD). Adapted from "Tools for Authoring a Dialogue Agent That Participates in Learning Studies," by P. W. Jordan, B. Hall, M. Ringenberg, Y. Cui, and C. P. Rosé in R. Lucklin, K. R. Koedinger, & J. E. Greer (Eds.), 2007. *AIED 2007: Proceedings of the 13th International Conference on Artificial Intelligence in Education, Los Angeles, CA* (p. 48). Copyright 2007 by IOS Press, Amsterdam, the Netherlands Adapted with permission.

Rule 8. *When the student uses a term incorrectly, give the definition of the term to help the student correct his or her mistake.*

This rule stems from the finding that the frequency of S–T elab(term:definition) relations, in which the tutor defined a term that the student stated incorrectly or misapplied, correlated with qualitative gains, particularly among high pretest students, $R(5) = .863$, $p = .012$. Example 16 illustrates this rule.

Rule 9. *The tutor should ask for missing units or prompt the student to provide them, especially when a student is performing well—for example, when the student is close to solving a problem or answering a qualitative question.*

This rule is based on the finding that the frequency of the aggregate variable object:attribute-units, which includes all exchanges in which the student presented a value without units and the tutor either provided these units or prompted the student to do so, correlated with qualitative learning among high pretest students, $R(5) = .817$, $p = .025$. In the following exchange, the tutor provides the missing units:

> *Example 21*
>
> *Student:* $T - mg = ma$; $500 - 539 = 55a$.
>
> *Tutor:* Good deal. (I would add units there by the way: $500N - 539N = 55\ \text{kg}^{*}a$.)

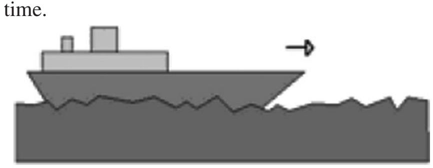
This rule is supported by prior research which used automated, machine learning methods to determine when abstractions and specification take place during reflective dialogues (Lipschultz, Litman, Jordon, & Katz, 2011). This research found that tutors tend to abstract over the student's dialogue contribution early in a reflective dialogue, when students are having difficulty responding to the tutoring system's reflection question. These abstractions appear to be aimed at ensuring that the student understands the basic concepts needed to answer the automated tutor's question. Then, as the dialogue progresses, and the student is closer to answering the reflection question correctly, specification becomes more frequent than abstraction, as tutors probe students for precision—for example, to specify units and direction for a vector quantity, when the student only states its magnitude.

An Example: Implementing Decision Rules in Rimac

Reflective dialogues in Rimac were developed using a well-known framework called *knowledge construction dialogues* (KCDs). Several NL tutoring systems that have been found to be more effective than noninteractive forms of instruction (e.g., reading a textbook) have been developed using the KCD framework (e.g., Aleven, Popescu, Ogan, & Koedinger, 2003; Connelly & Katz, 2009; Graesser et al., 2003; Graesser, Person, Harter, & the Tutoring Research Group, 2001; Jordan, Ringenberg, & Hall,

Table 8
Example of a Rimac Problem and Reflection Question

A modern supertanker is gigantic: 1,200–1,300 ft. long with a 200-ft. beam. Fully loaded, it chugs along at about 16.0 knots (8.33 m/s). It can take 1,200 s to bring it to a full stop. Calculate both the acceleration in m/s^2 and the displacement of the tanker using this stopping time.



Acceleration:
Displacement:

Reflection question: Is the acceleration pointing in the same direction or in the opposite direction of the supertanker's motion?

Note. This example problem is part of the Andes Physics Tutor system, which was developed at Arizona State University and the University of Pittsburgh with support from the Pittsburgh Science of Learning Center, National Science Foundation Award SBE-0836012, and Office of Naval Research Grant N00014-96-1-0260 and is available at http://www.andestutor.org

Table 9
Application of Rule 5 in an Experimental Knowledge Construction Dialogue (KCD) but Not in the Corresponding Control KCD

Experimental KCD	Control KCD
Tutor: In the problem you've just solved, is the acceleration pointing in the same direction or in the opposite direction of the supertanker's motion? *Student:* Opposite. *Tutor:* **So why did you think this?** *Student:* Because when an object is slowing down, acceleration is opposite to its motion. *Tutor:* Very good. Now let's step through the reasoning.	*Tutor:* In the problem you've just solved, is the acceleration pointing in the same direction or in the opposite direction of the supertanker's motion? *Student:* Opposite. *Tutor:* Very good. Now let's step through the reasoning.

Note. Bold typeface indicates application of the rule.

2006; Katz et al., 2007; Rosé et al., 2001). During a KCD, the automated tutor presents a series of carefully ordered questions to the student, known as a *directed line of reasoning* (DLR; Evens & Michael, 2006). If a student answers a question correctly, the student advances to the next question in the DLR. Otherwise, the system launches a remedial subdialogue and then returns to the main DLR after the remedial subdialogue has completed. This process is illustrated in Figure 1.

KCDs in Rimac were implemented using TuTalk, a NL-dialogue-authoring toolkit (Jordan et al., 2006; Jordan, Hall, Ringenberg, Cui, & Rosé, 2007). TuTalk enables domain experts to construct NL tutoring systems without programming. Instead, they can focus on defining the tutoring content and structure of KCDs.

From a research perspective, the main advantage of using KCDs is that the content and structure of KCDs are determined a priori by the dialogue developer, so different versions of a given KCD can be designed to test a hypothesis. Since our goal was to determine if the decision rules that we specified to guide simulation of cooperative execution during scaffolding enhance learning, we developed two versions of each Rimac KCD: one version that implements these rules in appropriate contexts and another that simulates the standard KCD practice of the tutor eliciting information from the student, hinting when possible, and stating the answer after the student has made one or two unsuccessful tries.

We illustrate these two versions of a Rimac KCD with respect to the problem and reflection question shown in Table 8. Dialogue excerpts illustrate implementation of three of the decision rules described in the preceding section in the experimental version of the dialogue.

In the dialogue excerpt shown in Table 9, the decision-rule-driven KCD applies Rule 5 because the student answered the question correctly but without justifying it: *The tutor should ask "why" questions when the student does not provide an explanation to support a claim, especially for less knowledgeable students.* In contrast, the standard KCD excerpt just gives the student positive feedback. In both cases, the dialogue steps through the line of reasoning that leads to the correct answer because even students who answer "why" questions correctly sometimes do so by rote.

An intermediate step of the line of reasoning is shown in Table 10. In this case, the experimental (decision-rule-driven) KCD applies Rule 6: *If the student answers a question incorrectly, if possible show why it is incorrect by stating the conditions under which it would be correct.* In contrast, the standard KCD simply corrects the student without acknowledging or addressing the student's response. Reformulating students' dialogue contributions, whether correct or incorrect, is a common practice during human tutoring, but its effectiveness has not been directly tested. We implemented specific procedures within TuTalk to simulate reformulation in Rimac (Jordan, Katz, Albacete, Ford, & Wilson, 2012).

At the final step of the line of line of reasoning, the decision-rule-driven KCD summarizes the main physics principle addressed in this reflection question (see Table 8) and recaps the line of reasoning that supports this principle, thereby applying Rule 1. In contrast, the standard KCD only states the general principle, as shown in Table 11.

These alternative versions of Rimac are currently undergoing pilot evaluation during the kinematics and dynamics units in high school physics classes. Several urban and suburban schools in the Pittsburgh, PA, area are participating in this pilot study. We predict

Table 10
Application of Rule 6 in an Experimental Knowledge Construction Dialogue (KCD) but Not in the Corresponding Control KCD

Experimental KCD	Control KCD
Tutor: When an object is slowing down, how does the final velocity (*vf*) compare to the initial velocity (*vi*) for any interval of time? (smaller, larger, other) *Student:* Larger *Tutor:* **If the object is speeding up, then its final velocity is larger than its initial velocity.** But when an object is slowing down, its velocity is getting smaller all the time. So for any interval of time, the final velocity is smaller than the initial velocity.	*Tutor:* When an object is slowing down, how does the final velocity (*vf*) compare to the initial velocity (vi) for any interval of time? (smaller, larger, other) *Student:* Larger *Tutor:* When an object is slowing down, its velocity is getting smaller all the time. So for any interval of time, the final velocity is smaller than the initial velocity.

Note. Bold typeface indicates application of the rule.

Table 11
Application of Rule 1 in an Experimental Knowledge Construction Dialogue (KCD) but Not in the Corresponding Control KCD

Experimental KCD	Control KCD
Tutor: Hence from the definition of acceleration, how do the direction of the acceleration and the direction of the change in velocity relate? (same, opposite, no relation, it depends) *Student:* They are the same. *Tutor:* Very good. **Hence, in the current problem, the direction of the acceleration opposes the direction of motion of the supertanker because the direction of the change in velocity opposes the motion of the supertanker. And this happens because the final velocity is smaller than the initial velocity, and velocity has the same direction as motion.** Therefore, in general, we can say that when an object is slowing down, the direction of the acceleration opposes the motion of the object.	*Tutor:* Hence from the definition of acceleration, how do the direction of the acceleration and the direction of the change in velocity relate? (same, opposite, no relation, it depends) *Student:* They are the same. *Tutor:* Very good. In general, when an object is slowing down, the direction of the acceleration opposes the motion of the object.

Note. Bold typeface indicates application of the rule.

that the decision-rule-driven version will outperform the less interactive control and that the effect will be greatest for less prepared students, a common finding for evaluations of instructional interventions (VanLehn et al., 2007).

Conclusion

The holy grail of tutoring research is to identify specific features of human tutorial dialogue that account for its remarkable effectiveness (e.g., Bloom, 1984; Cohen, Kulik, & Kulik, 1982), so that these features can be simulated in NL tutoring systems. Although the interaction hypothesis posits that more interactive tutoring will result in more learning, research to test this hypothesis shows that constructs like interactivity and cooperative execution are too vague to guide automated tutoring and, in particular, the scaffolding that takes place when students are having difficulty solving a quantitative problem or answering a conceptual question. In order to operationalize interactivity and cooperative execution, we need to identify the linguistic mechanisms that implement these constructs during human one-on-one tutoring and determine which mechanisms enhance learning. This knowledge can then be used to formulate decision rules that can be implemented and tested within NL tutoring systems. The research described in this article takes a step in this direction.

Overall, this study supports the interaction hypothesis. Our analyses suggest that the effectiveness of human tutoring might very well lie in the language of tutoring itself—in particular, in the types of discourse relations that students and tutors co-construct during tutorial dialogues. Moreover, the types of co-constructed discourse relations that predict learning seem to vary according to students' ability levels. However, given the small sample size, these findings should be cross-validated by analyses of dialogue corpora involving a larger number of subjects (both students and tutors).

A second limitation of this work stems from its focus on co-constructed discourse relations. It might well be the case that some discourse relations are better "told" than "elicited," that is, conveyed through direct, didactic explanations, instead of co-constructed while questioning the student. For example, we were surprised that we did not find a relationship between the frequency with which a tutor stated abstract principles or formulae (e.g., the equation for Newton's second law) and prompted students to instantiate these principles, as captured by the T–S abstract:instance relation, and student learning. However, this does not negate the potential effectiveness of instantiation of variables, principles, and so on during tutoring. Perhaps the didactic form of this relation (abstract:instance) does support learning, among some groups of students, but our analyses did not investigate correlations between didactically delivered discourse relations and learning. Hence, one goal of our future work will be to compare the effectiveness of didactic and interactive forms of particular discourse relations.

A third limitation of this research is that we did not consider variations in the way that co-construction of discourse relations is carried out and how these variations might impact learning. For example, we observed that there are two main ways in which tutors address abstractions. Tutors either anchor discussions about concepts and principles in the case at hand (i.e., the current problem) or address these abstractions in context-independent terms. For example, in both dialogue excerpts shown in Table 12, the tutor addresses the conditional: *if an object travels upward and comes back down, its vertical displacement is 0.* In the excerpt shown in the left column, the tutor grounds this abstraction in the current

Table 12
Alternative Ways of Prompting for a Conditional Relation

Context-specific prompt for a conditional relation	Context-independent prompt to complete a conditional relation
Tutor: Picture in your mind's eye . . . firecracker goes up, and then comes down and lands on the ground. What is the net vertical displacement for that whole process? *Student:* 0.	*Tutor:* Regardless of whether we call ground level $y = 0$ or $y = 500$, what is the y component of the displacement for an object that goes up and then comes back down to ground level? *Student:* 0 meters.

physical situation about a firecracker. He provides the antecedent of the conditional (the "if clause") and prompts the student for the consequent (the "then clause"). In contrast, in another dialogue about the same problem, shown in the right column of Table 12, the tutor speaks in more general, context-independent terms; he refers to "an object," not to the firecracker. Future research should examine which approach (if either) is better and for which types of students.

One important lesson that automated approaches to identifying decision rules to guide tutoring has taught us is that the "right" pedagogical move in a given context can depend on many factors: student characteristics, features of the problem under discussion, features of the dialogue context, and so on. We might not even be able to specify the relevant factors a priori. It is quite likely that we find that the decision rules suggested by our analyses are underspecified and in need of refinement. Although most of these rules, as stated, could apply to any scientific, problem-solving domain, their generalizability remains to be tested. A combination of automated approaches and carefully controlled, experimental studies of "tuned" versions of these decision rules and others will bring tutoring researchers closer to cracking the code of interactivity and developing more effective tutoring systems as a result.

References

Aleven, V., Popescu, O., Ogan, A., & Koedinger, K. (2003). A formative classroom evaluation of a tutorial dialogue system that supports self-explanation. In H. U. Hoppe, F. Verdejo, & J. Kay (Eds.), *AIED 2003: Proceedings of the 11th International Conference on Artificial Intelligence in Education, Sydney, Australia, July 2003* (pp. 39–46). Amsterdam, the Netherlands: IOS Press.

Beck, J., Woolf, B., & Beal, C. (2000). ADVISOR: A machine learning architecture for intelligent tutor construction. In H. Kautz & B. Porter (Co-chairs), *Proceedings of the Seventeenth National Conference on Artificial Intelligence, Austin, TX* (pp. 552–557). Menlo Park, CA: AAAI Press.

Bloom, B. S. (1984). The 2 sigma problem: The search for methods of group instruction as effective as one-to-one tutoring. *Educational Researcher, 13,* 4–16.

Boyer, K. E., Phillips, R., Ingram, A., Ha, E. Y., Wallis, M., Vouk, M., & Lester, J. (2010). Characterizing the effectiveness of tutorial dialogue with hidden Markov models. In V. Aleven, J. Kay, & J. Mostow (Eds.), *Intelligent tutoring systems: Proceedings of the 10th international conference on intelligent tutoring systems, ITS 2010, Pittsburgh, PA, June 14–18, 2010* (Pt. 1, Lecture Notes in Computer Science 6094, pp. 55–64). Berlin, Germany: Springer–Verlag.

Chi, M., Jordan, P., VanLehn, K., & Litman, D. (2009). To elicit or to tell: Does it matter? In V. Dimitrova, R. Mizoguchi, B. du Boulay, & A. Graesser (Eds.), *Proceedings of the 2009 Conference on Artificial Intelligence in Education. Building learning systems that care: From knowledge representation to affective modeling* (pp. 197–204). Amsterdam, the Netherlands: IOS Press.

Chi, M. T. H., Siler, S., Jeong, H., Yamauchi, T., & Hausmann, R. G. (2001). Learning from human tutoring. *Cognitive Science, 25,* 471–533. doi:10.1207/s15516709cog2504_1

Chi, M., VanLehn, K., Litman, D., & Jordan, P. (2010). Inducting effective pedagogical strategies using learning context features. In P. DeBra, A. Kobsa & D. Chin (Eds.), *User Modeling, Adaptation and Personalization: 18th International Conference, UMAP 2010* (pp. 147–158). Heidelberg, Germany: Springer.

Chi, M., VanLehn, K., Litman, D., & Jordan, P. (2011a). An evaluation of pedagogical tutorial tactics for a natural language tutoring system: A reinforcement learning approach. *International Journal of Artificial Intelligence in Education, 21,* 83–113.

Chi, M., VanLehn, K., Litman, D., & Jordan, P. (2011b). Empirically evaluating the application of reinforcement learning to the induction of effective and adaptive pedagogical strategies. *User Modeling and User-Adapted Interaction, 21,* 137–180.

Clark, H. H., & Schaefer, E. F. (1989). Contributing to discourse. *Cognitive Science, 13,* 259–294. doi:10.1207/s15516709cog1302_7

Cohen, P. A., Kulik, J. A., & Kulik, C.-L. C. (1982). Educational outcomes of tutoring: A meta-analysis of findings. *American Educational Research Journal, 19,* 237–248.

Collins, A., & Brown, J. S. (1986). *The computer as a tool for learning through reflection* (Technical Rept. 376). Washington, DC: National Institute of Education.

Connelly, J., & Katz, S. (2009). Towards more robust learning of physics via reflective dialogue extensions. In C. Fulford (Ed.), *ED-MEDIA 2009: World Conference on Educational Multimedia, Hypermedia, & Telecommunications, Honolulu, Hawaii, June 22–26, 2009*. Chesapeake, VA: Association for the Advancement of Computing in Education.

Di Eugenio, B., & Green, N. L. (2010). Emerging applications of natural language generation in information visualization, education, and health care. In N. Indurkhya & F. J. Damerau (Eds.), *Handbook of natural language processing* (2nd ed.; pp. 557–576). Boca Raton, FL: Chapman & Hall/CRC.

Evens, M. W., & Michael, J. A. (2006). *One-on-one tutoring by humans and machines*. Mahwah, NJ: Erlbaum.

Garrod, S., & Pickering, M. J. (2004). Why is conversation so easy? *Trends in Cognitive Sciences, 8,* 8–11. doi:10.1016/j.tics.2003.10.016

Gick, M., & Holyoak, K. (1983). Schema induction and analogical transfer. *Cognitive Psychology, 15,* 1–38. doi:10.1016/0010-0285(83)90002-6

Gick, M., & Holyoak, K. (1987). The cognitive basis of knowledge transfer. In S. M. Cormier & J. D. Hagman (Eds.), *Transfer of learning: Contemporary research and applications* (pp. 9–46). New York, NY: Academic Press.

Graesser, A. C., Jackson, G. T., Mathews, E. C., Mitchell, H. H., Olney, A., Ventura, M., . . . Person, N. K. (2003). Why/AutoTutor: A test of learning gains from a physics tutor with natural language dialogue. In R. Alterman & D. Kirsh (Eds.), *Proceedings of the 25th Annual Conference of the Cognitive Science Society 2003* (pp. 474–479). Boston, MA: Cognitive Science Society.

Graesser, A. C., Person, N. K., Harter, D., & the Tutoring Research Group. (2001). Teaching tactics and dialog in AutoTutor. *International Journal of Artificial Intelligence in Education, 12,* 257–279.

Graesser, A. C., Person, N. K., & Magliano, J. P. (1995). Collaborative dialogue patterns in naturalistic one-on-one tutoring. *Applied Cognitive Psychology, 9,* 495–522. doi:10.1002/acp.2350090604

Halliday, M. A. K., & Hasan, R. (1976). *Cohesion in English* (English Language Series). London, England: Pearson Education.

Halloun, I., & Hestenes, D. (1985). The initial knowledge state of college physics students. *American Journal of Physics, 53,* 1043–1055. doi:10.1119/1.14030

Hovy, E. (1990). Parsimonious and profligate approaches to the question of discourse structure relations. In K. R. McKeown, J. D. Moore, & S. Nirenburg (Eds.), *Proceedings of the Fifth International Workshop on Natural Language Generation,* June 3–6, 1990, *Dawson, PA* (pp. 59–65). Stroudsburg, PA: Association for Computational Linguistics Special Interest Group on Natural Language Generation (SIGGEN).

Jordan, P. W., Hall, B., Ringenberg, M., Cui, Y., & Rosé, C. P. (2007). Tools for authoring a dialogue agent that participates in learning studies. In R. Lucklin, K. R. Koedinger, & J. E. Greer (Eds.), *AIED 2007: Proceedings of the 13th International Conference on Artificial Intelligence in Education, Los Angeles, CA* (pp. 43–50). Amsterdam, the Netherlands: IOS Press.

Jordan, P. W., Katz, S., Albacete, P., Ford, M., & Wilson, C. (2012).

Reformulating student contributions in tutorial dialogue. In B. Di Eugenio, S. McRoy, A. Gatt, A. Betz, A. Koller, & K. Striegnitz (Eds.), *INGL 2012: Proceedings of 7th International Natural Language Generation Conference, Utica, IL, May 30–June 1, 2012* (pp. 95–99). Stroudsburg, PA: Association for Computational Linguistics Special Interest Group on Natural Language Generation (SIGGEN).

Jordan, P. W., Ringenberg, M., & Hall, B. (2006). Rapidly developing dialogue systems that support learning studies. In E. Lulis & P. Wiemer-Hastings (Eds.), *Proceedings of ITS 2006 Workshop on Teaching with Robots, Agents, and NLP*. Jhongli, Taiwan: National Center University Research Center for Science and Technology for Learning.

Katz, S., Allbritton, D., & Connelly, J. (2003). Going beyond the problem given: How human tutors use post-solution discussions to support transfer. *International Journal of Artificial Intelligence and Education, 13,* 79–116.

Katz, S., Connelly, J., & Wilson, C. (2007). Out of the lab and into the classroom: An evaluation of reflective dialogue in Andes. In R. Lucklin, K. R. Koedinger, & J. E. Greer (Eds.), *AIED 2007: Proceedings of the 13th International Conference on Artificial Intelligence in Education, Los Angeles, CA* (pp. 425–432). Amsterdam, the Netherlands: IOS Press.

Lee, A., & Hutchison, L. Improving learning from examples through reflection. *Journal of Experimental Psychology: Applied, 4,* 187–210.

Leher, R., & Littlefield, J. (1993). Relationships among cognitive components in logo learning and transfer. *Journal of Educational Psychology, 85,* 317–330. doi:10.1037/0022-0663.85.2.317

Lipschultz, M., Litman, D., Jordan, P., & Katz, S. (2011). Predicting changes in level of abstraction in tutor responses to students. In R. C. Murray & R. M. McCarthy (Eds.), *Proceedings of the 24th International Florida Artificial Intelligence Research Society Conference (FLAIRS 2011), May 18–20, 2011, Palm Beach, FL*. Miami, FL: FLAIRS.

Litman, D., & Forbes-Riley, K. (2006). Correlations between dialogue acts and learning in spoken tutoring dialogues. *Natural Language Engineering, 12,* 161–176. doi:10.1017/S1351324906004165

Louwerse, M. M., Crossley, S. A., & Jeuniaux, P. (2008). What if? Conditionals in educational registers. *Linguistics and Education, 19,* 56–69. doi:10.1016/j.linged.2008.01.001

Mann, W. C., & Thompson, S. (1988). Rhetorical structure theory: Toward a functional theory of text organization. *Text, 8,* 243–281. doi:10.1515/text.1.1988.8.3.243

McNamara, D. S. (2001). Reading both high-coherence and low-coherence texts: Effects of text sequence and prior knowledge. *Canadian Journal of Experimental Psychology, 55,* 51–62.

McNamara, D. S., & Kintsch, W. (1996). Learning from text: Effects of prior knowledge and text coherence. *Discourse Processes, 22,* 247–287. doi:10.1080/01638539609544975

McNamara, D. S., Kintsch, E., Songer, N. B., & Kintsch, W. (1996). Are good texts always better? Text coherence, background knowledge, and levels of understanding in learning from text. *Cognition and Instruction, 14,* 1–43. doi:10.1207/s1532690xci1401_1

Murray, R. C., & VanLehn, K. (2006). A comparison of decision-theoretic, fixed-policy and random tutorial action selection. In K. Ashley & M. Ikeda (Eds.), *Intelligent tutoring systems: Proceedings of the eighth international conference, ITS 2006, Jhongli, Taiwan, June 26–30, 2006* (Lecture Notes in Computer Science 4053, pp. 114–123). Berlin, Germany: Springer–Verlag. doi:10.1007/11774303_12

Obama, B. (2009). *Remarks by the president at the National Academy of Sciences Annual Meeting*. Retrieved March 13, 2013, from http://www.whitehouse.gov/the_press_office/Remarks-by-the-President-at-the-National-Academy-of-Sciences-Annual-Meeting

Pilkington, R. (2001). Analysing educational dialogue interaction: Towards models that support learning. *International Journal of Artificial Intelligence in Education, 12,* 1–7.

Ravenscroft, A., & Pilkington, R. M. (2000). Investigation by design: Developing models to support reasoning and conceptual change. *International Journal of Artificial Intelligence in Education, 11,* 273–298.

Reed, S. K. (1993). A schema-based theory of transfer. In D. K. Detterman & R. J. Sternberg (Eds.), *Transfer on trial: Intelligence, Cognition, and instruction* (pp. 39–67). Norwood, NJ: Ablex.

Rosé, C., Jordan, P., Ringenberg, M., Siler, S., VanLehn, K., & Weinstein, A. (2001). Interactive conceptual tutoring in Atlas–Andes. In J. D. Moore, C. L. Redfield, & W. L. Johnson (Eds.), *Artificial intelligence in education* (pp. 256–266). Amsterdam, the Netherlands: IOS Press.

Salomon, G., & Perkins, D. N. (1989). Rocky roads to transfer: Rethinking mechanisms of a neglected phenomenon. *Educational Psychologist, 24,* 113–142. doi:10.1207/s15326985ep2402_1

Steinhauser, N., Campbell, G. E., Taylor, L. S., Caine, S., Scott, C., Dzikovska, M., & Moore, J. D. (2011). Talk like an electrician: Mimicking behavior in an intelligent tutoring system. In G. Biswas, S. Bull, J. Kay, & A. Mitrovic (Eds.), *Artificial intelligence in education: 15th international conference, AIED 2011, Auckland, New Zealand, June 28–July 2, 2011* (Lecture Notes in Artificial Intelligence 6738, pp. 361–368). Berlin, Germany: Springer.

Tchetagni, J. M. P., Nkambou, R., & Bourdeau, J. (2007). Explicit reflection in prolog-tutor. *International Journal of Artificial Intelligence in Education, 17,* 169–215.

van de Sande, C., & Greeno, J. G. (2010). A framing of instructional explanations: Let us explain with you. In M. K. Stein & L. Kucan (Eds.), *Instructional explanations in the disciplines* (pp. 69–82). New York, NY: Springer.

VanLehn, K. (2011). The relative effectiveness of human tutoring, intelligent tutoring systems, and other tutoring systems. *Educational Psychologist, 46,* 197–221. doi:10.1080/00461520.2011.611369

VanLehn, K., Graesser, A. C., Jackson, G. T., Jordan, P., Olney, A., & Rosé, C. P. (2007). When are tutorial dialogues more effective than reading? *Cognitive Science, 31,* 3–62. doi:10.1080/03640210709336984

VanLehn, K., Lynch, C., Schulze, K., Shapiro, J. A., Shelby, R., Taylor, L., . . . Wintersgill, M. (2005). The Andes physics tutoring system: Lessons learned. *International Journal of Artificial Intelligence in Education, 15,* 1–47.

Ward, A., Connelly, J., Katz, S., Litman, D., & Wilson, C. (2009). Cohesion, semantics, and learning in reflective dialog. In S. D. Craig & D. Dicheva, *AIED 2009: 14th International Conference on Artificial Intelligence in Education Workshop Proceedings. Vol. 10: Natural Language Processing in Support of Learning. Metrics, Feedback, and Connectivity*. Available at http://webu2.upmf-grenoable.fr/sciedu/nlpsl

Ward, A., & Litman, D. (2008). Semantic cohesion and learning. In B. Woolf, E. Aimeur, R. Nkambou, & S. Lajoie (Eds.), *Intelligent tutoring systems: Proceedings of the 9th International Conference, ITS 2008, Montreal, Canada, June 23–27, 2008* (Lecture Notes in Computer Science 5091, pp. 459–469). New York, NY: Springer.

Ward, A., & Litman, D. (2011). Adding abstractive reflection to a tutorial dialog system. In R. C. Murray & R. M. McCarthy (Eds.), *Proceedings of the 24th International Florida Artificial Intelligence Research Society Conference (FLAIRS 2011), May 18–20, 2011, Palm Beach, FL* (Paper 2575). Miami, FL: FLAIRS.

Received December 15, 2011
Revision received October 22, 2012
Accepted December 18, 2012 ■